REMARKS

Applicant thanks Examiner for the detailed review of the application. Applicant has amended claims 1, 7, 10, 11, 12, 18, 26-31, and 33, cancelled claims 5, 8, and 32, and added new claims 34-41. Examiner currently rejects claim 7 under 35 U.S.C. 112, second paragraph, as being indefinite. Applicant has amended claim from "the execution pipeline" to "an execution pipeline" to provide adequate antecedent basis.

Examiner currently rejects applicant's claims 1-3 and 6-7 under 35 U.S.C. 101 because the claimed invention is directed to non-statutory subject matter. Applicant has amended claim 1 to "a processor including," which is a useful, concrete, and tangible result, i.e. a processor with marking and blocker logic. Therefore, applicant respectfully requests that claims 1-3 and 6-7 pass the Practical Test application and are now in condition for allowance.

Examiner currently rejects claims 1-33 under 35 U.S.C. 102(e) as being anticipated by Liao et al. (US 2004/0054990), herein referred to a Liao.

Applicant's claim 1 includes the element, "blocker logic to prevent data associated with a store instruction of the speculative thread from being forwarded to an instruction of a non-speculative thread, the blocker logic further to prevent the data from being stored in a memory system." In contrast, Liao discloses the following at paragraph 0045:

> [0045] At block **612**, the method **100** ensures that no store instructions are included in the slice. One skilled in the art will recognize that exclusion of store instructions need not occur temporally after the region-based and context-sensitive slicing, and can be performed in any order with respect to blocks **608** and **610**.

As can be seen, Liao only suggests excluding store instructions themselves from a pre-computed speculative slice. As a result, a compiler provides the rules to exclude store instructions themselves. However, applicant's claim 1 includes **blocker logic** to prevent **data associated with a store**

instruction of the speculative thread from being forwarded. In direct contrast to Liao, a store instruction may be included in the speculative thread slice and executed; however, applicant's claim 1 blocker logic is to prevent **data associated** with the store instruction from being forwarded, not the exclusion of all store instructions themselves from speculative thread slices with a compiler, as in Liao.

Similarly, applicant's claim 11 includes the element, "dependence blocker logic to prevent data associated with a store instruction of a speculative thread from being forwarded to an instruction of a non-speculative thread." As stated above, Liao only suggests use of a compiler to exclude all store instructions from a pre-computed speculative slice, not logic available at runtime to prevent data associated with store instructions included in a speculative slice from being forwarded to a non-speculative thread.

In addition, applicant's claim 21 includes the elements, "determining if the load instruction and the **in-flight store instruction** each **originate with a speculative thread**" As above, all store instructions are explicitly excluded from Liao's speculative pre-computed slices. However, applicant's claim 21 includes an in flight store instruction, which may have originated from a speculative thread, which Liao directly teaches away from.

Liao only discloses fetching an address from memory (or a higher-level cache) into a lower memory level, and does not disclose or suggest any of applicant's elements in amended claim 26, which are to be performed in response to a speculative thread cache write request.

Furthermore, applicant's newly added claim 34 includes a storage area to include a speculation identifier (ID) field. As stated above, Liao only discloses operation of a compiler to pre-compute a speculative slice, not a storage area include a speculation ID field, as disclosed in applicant's claim 34.

Consequently, applicant respectfully submits that independent claims 1, 11, 21, 26, and 34, as well as there dependent claims, are now in condition for allowance for at least the reasons stated above.

If there are any additional charges, please charge Deposit Account No. 50-0221. If a telephone interview would in any way expedite the prosecution of the present application, the Examiner is invited to contact David P. McAbee at (503) 712-4988.

Respectfully submitted,
Intel Corporation

Dated: April 30, 2007

/David P. McAbee/Reg. No. 58,104/
David P. McAbee
Reg. No. 58,104

Intel Corporation
M/S JF3-147
2111 NE 25th Avenue
Hillsboro, OR 97124
Tele – 503-712-4988
Fax – 503-264-1729